TU München, Fakultät für Informatik
Lehrstuhl III: Datenbanksysteme
Prof. Alfons Kemper, Ph.D.

Übung zur Vorlesung *Grundlagen: Datenbanken* im WS22/23

Michael Jungmair, Stefan Lehner, Moritz Sichert, Lukas Vogel (gdb@in.tum.de)
https://db.in.tum.de/teaching/ws2223/grundlagen/

Blatt Nr. 01

Das Uni-Schema finden Sie auf Seite 20 des zweiten Foliensatzes: https://db.in.tum.de/teaching/bookDBMSeinf/folien/pdf/Kapitel2.pdf.

Hausaufgabe 1

Schätzen Sie die Größe der Datenbank des Amazon-Shops ab. Berücksichtigen Sie dabei Produkte, Kunden und Bestellungen. Schätzen Sie außerdem ab, wie viele Transaktionen pro Sekunde von dieser Datenbank abgewickelt werden und leiten Sie daraus ab, wie schnell die Datenbank wächst (Bytes pro Sekunde).

a) Ermitteln Sie möglichst aktuelle statistische Werte für die Anzahl der Produkte, Kunden und Bestellungen sowie die durchschnittliche Anzahl neuer Bestellungen pro Sekunde.

b) Berechnen Sie anhand der in a) geschätzten Werte die Größe der Datenbank und den durch neuen Bestellungen verursachten Durchsatz (Bytes pro Sekunde).

Lösung:

a) Ermitteln Sie möglichst aktuelle statistische Werte für die Anzahl der Produkte, Kunden und Bestellungen sowie die durchschnittliche Anzahl neuer Bestellungen pro Sekunde.

Geschätzte Werte (Ihre Lösung kann natürlich stark von unserer abweichen, da es keine verlässlichen Zahlen dazu gibt):

- 350 Millionen verschiedene Produkte
- 600 Millionen Kunden
- 1,5 Milliarden Bestellungen
- 20 Bestellungen pro Sekunde

b) Berechnen Sie anhand der in a) geschätzten Werte die Größe der Datenbank und den durch neuen Bestellungen verursachten Durchsatz (Bytes pro Sekunde).

- Größe eines Datensatzes eines Produkts:
  - Beschreibung + Name (1 KB)
  - Produktbilder (1 MB)
  - Produktvideos (1 MB durchschnittlich, nicht alle Produkte haben ein Video)
  - Preis(e), Varianten und Lieferbedingungen (Etwa 1 KB)
  - Rezensionen potentiell mit Bildern und Videos (5 MB)

Insgesamt also 7,002 MB pro Produkt.

- Größe eines Datensatzes eines Kunden:
  - Stammdaten (Name, E-Mail-Adresse, etc.) (1 KB)
  - Zahlungs- und Adressinformationen (1 KB)

  Insgesamt also 2 KB pro Kunde.
- Größe einer Bestellung:
  - Referenz auf den Kunden (8 B)
  - Pro bestelltem Artikel:
    * Referenz auf Artikel (8 B)
    * Anzahl (8 B)
    * Referenz auf Lieferant (8 B)
    * Lieferdatum (8 B)
    * Lieferhinweise oder -kommentare (50 B)
    * Möglicher Rabatt (8 B)
    * Steuer (8 B)

    Pro Artikel also 98 B $\approx$ 100 B. Bei durchschnittlich 2 Artikeln pro Bestellung also 200 B.

Für die Gesamtgrößen ergibt sich dann, wenn man die Werte aus a) nimmt:

- Produkte: $350 \cdot 10^6 \cdot 7,002$ MB $= 2,4507$ PB
- Kunden: $600 \cdot 10^6 \cdot 2$ KB $= 1,2$ TB
- Bestellungen: $1,5 \cdot 10^9 \cdot 200$ B $= 300$ GB

Der Durchsatz der Bestellungen ist $20 \cdot 200$ B/s $= 4$ KB/s.

Hausaufgabe 2

Sie designen eine Webanwendung zur Univerwaltung. Früh entschließen Sie sich zum Einsatz eines Datenbanksystems als Backend für Ihre Daten. Ihr Kollege ist skeptisch und würde die Datenverwaltung lieber selbst implementieren. Überzeugen Sie ihn von Ihrem Entschluss. Finden Sie stichhaltige Antworten auf die folgenden von Ihrem Kollegen in den Raum gestellten Äußerungen:

a) Die Installation und Wartung eines Datenbanksystems ist aufwendig, die Erstellung eines eigenen Datenformats ist straight-forward und flexibler.

b) Mehrbenutzersynchronisation wird in diesem Fall nicht benötigt.

c) Es ist unsinnig, das jeder Entwickler zunächst eine eigene Anfragesprache (SQL) lernen muss, nur um Daten aus der Datenbank zu extrahieren.

d) Redundanz ist hilfreich, wieso sollte man auf sie verzichten?

Lösung:

a) Einige Datenformate sind inhärent unflexibel, da es keinen standardisierten Pfad zur Erweiterung, Verteilung, Recovery etc. gibt. Man bedenke beispielsweise, dass allein viele Dateisysteme keine Dateien über einer fixen Größe, bei FAT32 beispielsweise

traditionell 2GB erlauben. Hinzu kommt, dass durch die manuelle Erstellung von Dateiformaten keine standardisierten Datentypen verwendet werden und das gesamte "Schema" der Datenspeicherung leicht uneinheitlich wird. Im Vergleich dazu ist der Aufwand zu Erstinstallation einer Datenbank vernachlässigbar, insbesondere, da heutzutage nicht zwangsläufig ein komplexes Produkt wie IBM DB2[1] o.Ä. verwendet werden muss sondern man für kleine Eigenentwicklungen auch durchaus eine eingebettete Datenbank wie etwa sqlite[2] verwendet werden kann.

b) Mehrbenutzersynchronisation ist inhärent notwendig, wenn sie ein System entwickeln, auf das mehrere Personen zugreifen. Insbesondere ist diese eine der Eigenschaften, die nicht einfach "nachgepatched" werden kann, sondern sehr tief in eine Datenverwaltungsschicht integriert werden muss. Datenbanksysteme erlauben es, ohne über Nebenläufigkeit nachdenken zu müssen auf Daten zuzugreifen und das "erwartete" Ergebnis zu erhalten. Siehe dazu das ACID Paradigma[3], was i.A. von DBMS erfüllt wird.

c) Ein eigenes Datenformat und dessen API (wenn es denn zumindest eine API für den Zugriff gibt) muss auch gelernt werden, dafür ist SQL standardisiert und kann auch beim Wechsel des DBMS weiterverwendet werden.

d) Redundanz sorgt auch für Anomalien, etwa beim Updaten von Daten.

Hausaufgabe 3

Beim konzeptuellen Entwurf hat man gewisse Freiheitsgrade hinsichtlich der Modellierung der realen Welt. Unter anderem hat man folgende Alternativen, die Sie an unserem Universitätsschema beispielhaft illustrieren sollten:

- Man kann ternäre Beziehungen in binäre Beziehungen transformieren.

  Betrachten Sie dazu die Beziehung *prüfen* und erläutern Sie die Vor- und Nachteile einer solchen Transformation.

- Man hat manchmal die Wahl, ein Konzept der realen Welt als Beziehung oder als Entitytyp zu modellieren. Erörtern Sie dies wiederum am Beispiel der Beziehung *prüfen* im Gegensatz zu einem eigenständigen Entitytyp *Prüfungen*.

- Ein Konzept der realen Welt kann manchmal als Entitytyp mit zugehörigem Beziehungstyp und manchmal als Attribut dargestellt werden. Ein Beispiel hierfür ist das Attribut *Raum* des Entitytyps *Professoren* im bekannten Uni Schema. Diskutieren Sie die Alternativen.

Lösung:

Ziel dieser Aufgabe ist es, alternative Entwürfe zu erstellen und bezüglich ihrer Anwendbarkeit zu analysieren. Unter Anwendbarkeit ist unter anderem zu verstehen, ob in der neuen Modellierung dieselben Informationseinheiten wie in der ursprünglichen abgebildet werden können, ob Konsistenzbedingungen eingehalten werden und ob die reale Welt in der modellierten Miniwelt sinnvoll wiedergegeben ist.

[1] `http://www-01.ibm.com/software/data/db2/`
[2] `http://www.sqlite.org/`
[3] `http://en.wikipedia.org/wiki/ACID`

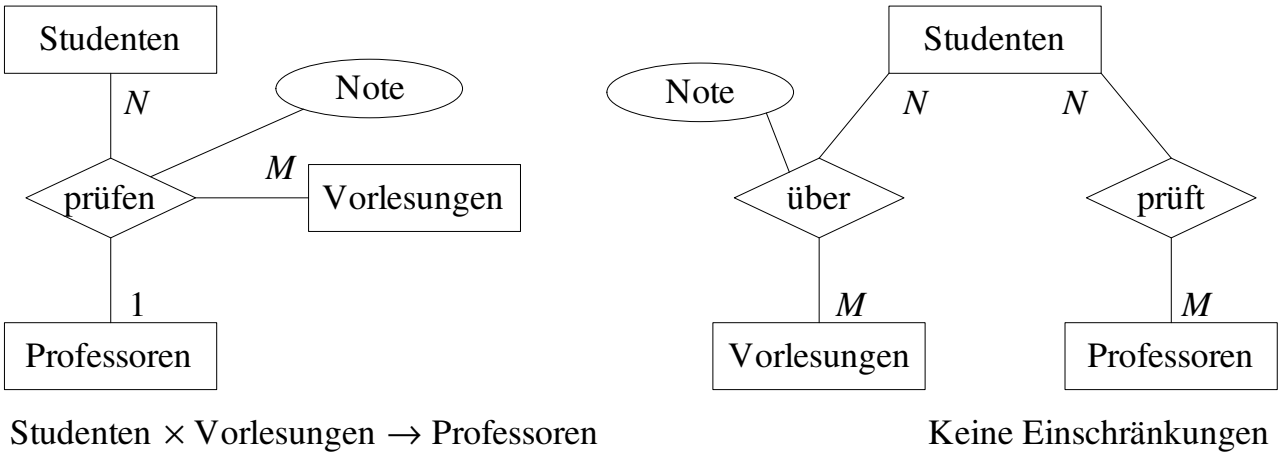


Abbildung 1: Auflösen der ternären Beziehung *prüfen* in binäre Beziehungen

Erste Teilaufgabe: Transformation der ternären Beziehung in binäre Beziehungen

Abbildung 1 zeigt einen Ansatz, die ternäre Beziehung *prüfen* durch binäre Relationen auszudrücken. Durch die ursprüngliche Modellierung (links in der Abbildung) wird folgende Konsistenzbedingung ausgedrückt:

$$\text{prüfen} : \text{Studenten} \times \text{Vorlesungen} \quad \rightarrow \quad \text{Professoren} \tag{1}$$

Demgegenüber tritt bei der vorgeschlagenen Modellierung mittels binärer Relationen ein Semantikverlust auf. Durch die allgemeineren N:M-Beziehungen wird obige Konsistenzbedingung nicht mehr abgebildet. Somit ist das Modell der ternären Beziehung in diesem Fall ausdrucksstärker. Zwar lassen sich Prüfungsergebnisse in der alternativen Modellierung abbilden, allerdings geht Aussagekraft verloren. Abgebildet ist, dass Studenten über den Stoff von Vorlesungen geprüft werden, sowie dass Studenten von Professoren geprüft werden. Der Zusammenhang, welche Professoren welche Studenten in welchen Vorlesungen prüfen, ist aber nicht mehr ohne weiteres gegeben. Indirekt lösen lässt sich dies durch die Aufnahme des zusätzlichen Attributs *Prüfungszeit* in die Relation *über* und auch in *prüft*. Da der zusätzlich aufgeführte Prüfungstermin eine Prüfung eindeutig festlegt, lässt sich die Information über eine Prüfung aus beiden Relationen erhalten. Allerdings muss für eine konsistente Extension sichergestellt werden, dass zu einem Eintrag in *über* auch ein passender Eintrag in *prüft* enthalten ist. Die gezeigte alternative Modellierung weist also klare Nachteile gegenüber der ursprünglichen ternären Beziehung auf.

Die alternative Modellierung einer ternären Beziehung durch mehrere binäre kann (abhängig von den zu modellierenden Anforderungen) im Allgemeinen folgende Nachteile aufweisen:

- Es tritt ein Semantikverlust auf.
- Es besteht die Möglichkeit, inkonsistente Datenbankzustände zu generieren. Gegebenenfalls ist eine Konsistenzüberprüfung der Datenbank erforderlich.
- Die reale Welt wird in der Miniwelt unzureichend wiedergegeben.

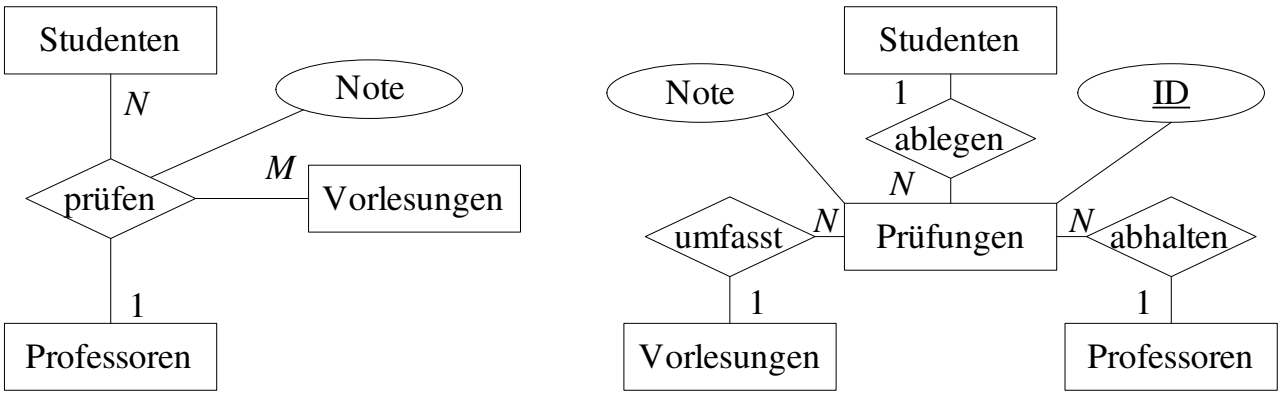


Abbildung 2: Modellierung von Prüfungen als Entitytyp

Zweite Teilaufgabe: Modellierung als Entitytyp anstelle einer Beziehung

Abbildung 2 zeigt eine alternative Modellierung der *prüfen*-Beziehung über einen Entitytyp *Prüfungen*. Auch in diesem Fall tritt erneut ein Semantikverlust auf. Es ist möglich, dass in der Modellierung mittels Entitytyp eine Prüfung existiert, zu der z.B. noch kein Prüfer feststeht, bzw. der Prüfer nicht mehr existiert. Möchte man dies in der Modellierung ausdrücken, müsste man zur (*min*,*max*)-Notation übergehen, mittels derer man fordern kann, dass eine Prüfung genau je einmal in den Relationen *ablegen*, *abhalten* und *umfasst* auftritt. Außerdem kann auch obige Konsistenzbedingung (1) nicht zugesichert werden. Zwar legt eine *Prüfungen*-Instanz über die angesprochenen Relationen den Studenten / die Studentin, die geprüfte Vorlesung und den / die prüfende(n) Professor(in) fest. Allerdings bestimmt die Kombination *Studenten* × *Vorlesungen* nun nicht mehr *Professoren*. Denn es ist durch den Entwurf nicht ausgeschlossen, dass es beispielsweise zwei unterschiedliche Prüfungen gibt, die der Student Fichte im Fach Ethik ablegt. Nur einmal lässt er sich bei Professor Sokrates und ein andermal bei Professorin Curie darüber prüfen. Andererseits lassen sich manche Aspekte der Modellierung mittels Entity genauer erfassen, als dies in der ursprünglichen Modellierung der Fall ist. So ist in obigem Beispiel etwa spezifiziert, dass pro Prüfung genau eine Vorlesung geprüft wird.

Dritte Teilaufgabe: Alternative Modellierungen über Attribute oder Beziehungstypen

Abbildung 3 zeigt zwei Modellierungsansätze, die ausdrücken, dass jedem Professor ein Raum zugewiesen ist. Links die Darstellung mittels Attribut, rechts über die Beziehung *residiertIn*. Generell ist eine Modellierung über eine Beziehung mit einem eigenständigen Entity (*Raum*) dann angebracht, wenn entsprechend detaillierte Informationen zu einem Raum nötig sind. Dies kann z.B. dann der Fall sein, wenn die Anwendungssicht der Abteilung Gebäudetechnik in das Modell integriert werden muss. Möchte man die Raumdaten für jeden Professor abfragen, dann zieht diese Modellierung in der Regel eine weniger effiziente Anfrageauswertung nach sich.

Hausaufgabe 4

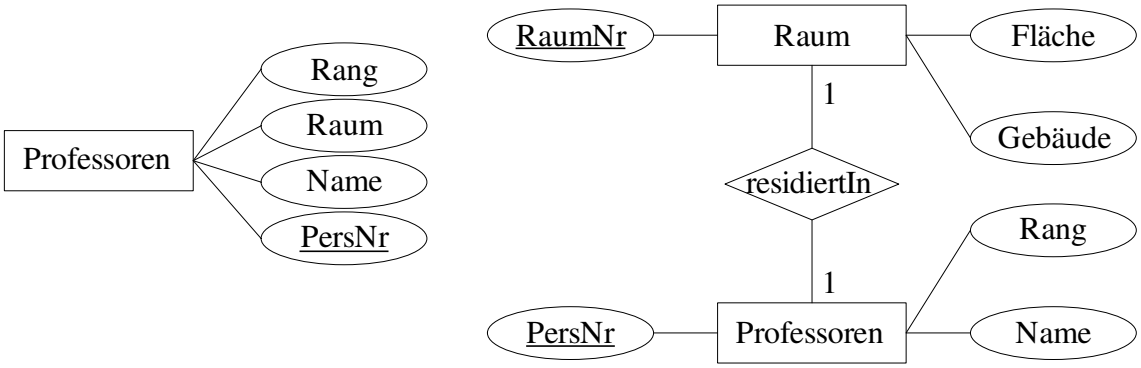
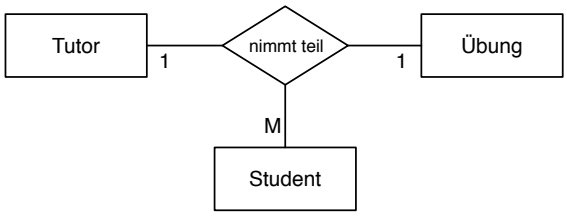


Abbildung 3: Zwei alternative Modellierungen um auszudrücken, dass Professoren ein Raum zugeordnet ist



Ignorieren Sie die Funktionalitätsangaben und beantworten Sie:

a) Wie viele partielle Funktionen der Form $A \times B \rightarrow C$ können in einer ternären Beziehung auftreten (Ignorieren Sie beim Zählen die Reihenfolge auf der linken Seite der Beziehung).

b) Nennen Sie alle möglichen partiellen Beziehungen in der hier gezeigten Beziehung „nimmt teil“.

c) Nennen Sie für jede Funktion in Prosa, welche Einschränkung diese darstellt, falls sie gilt.

Unter Berücksichtigung der Funktionalitätsangaben:

d) Welche partiellen Funktionen gelten hier?

Lösung:

a) Es gibt drei mögliche partielle Funktionen

b)

$$Tutor \times Uebung \rightarrow Student \quad (2)$$
$$Tutor \times Student \rightarrow Uebung \quad (3)$$
$$Uebung \times Student \rightarrow Tutor \quad (4)$$

c) Würde Funktion 2 gelten, so darf ein Tutor pro Übung nur einen Studenten haben. Gilt Funktion 3, so darf ein Student bei einem Tutor nur eine Übung besuchen. Gilt Funktion 4, so darf es für einen konkreten Studenten in einer Übung nur einen Tutor geben.

d) Zu den in der Abbildung gezeigten Kardinalitätsangaben „passen“ die partiellen Funktionen 3 und 4, weshalb diese für das Beispiel gelten. 2 gilt hingegen - wie auch bei uns im Übungsbetrieb - nicht.